A TEST-BED FOR QUALITY OF MULTIMEDIA EXPERIENCE EVALUATION OF SENSORY EFFECTS

Markus Waltl, Christian Timmerer, and Hermann Hellwagner

Multimedia Communication (MMC) Research Group, Institute of Information Technology (ITEC)
Klagenfurt University, Klagenfurt, Austria
E-mail: *firstname.lastname*@itec.uni-klu.ac.at

ABSTRACT

This paper introduces a prototype test-bed for triggering sensory effects like light, wind, or vibration when presenting audiovisual resources, e.g., a video, to users. The ISO/IEC MPEG is currently standardizing the Sensory Effect Description Language (SEDL) for describing such effects. This language is briefly described in the paper and the test-bed that is destined to evaluate the quality of the multimedia experience of users is presented. It consists of a video annotation tool for sensory effects, a corresponding simulation tool, and a real test system. Initial experiments and results on determining the color of light effects from the video content are reported.

***Index Terms*—**Quality of Multimedia Experience, Sensory Effects, Sensory Effect Description Language, SEDL, MPEG

1 INTRODUCTION

The usage of multimedia content is becoming omnipresent in everyday life, in terms of both consumption and production. On the one hand, professional content is provided to the end user in high-definition quality, streamed over heterogeneous networks, and consumed on a variety of different devices. On the other hand, user-generated content overwhelms the Internet with multimedia assets being uploaded to a wide range of available Web sites. That is, the transparent access to multimedia content – also referred to as Universal Multimedia Access (UMA) [1] – seems to be technically feasible. However, UMA mainly focuses on the end-user devices and network connectivity issues, but it is the user who ultimately consumes the content. Hence, the concept of UMA has been extended to take the user into account, which is generally referred to as Universal Multimedia Experience (UME) [2].

The research efforts around UMA/UME resulted in many publications in the areas of multimedia adaptation (e.g., [1]) and multimedia quality models (e.g., [3]). However, most of these quality models are restricted to a single modality (i.e., audio, image, or video only) or a simple combination of two modalities (i.e., audio and video). In [4] a triple user characterization model for video adaptation and Quality of Experience (QoE) evaluation is described that introduces at least three quality evaluation dimensions, namely sensorial (e.g., sharpness, brightness), perceptual (e.g., what/where is the content), and emotional (e.g., feeling, sensation) evaluation. Furthermore, it proposes adaptation techniques for the multimedia content and quality metrics associated to each of these layers. The focus is clearly on how an audio/visual resource is perceived, possibly taking into account certain user characteristics (e.g., handicaps) or natural environment conditions (e.g., illumination). In [5] the authors report – based on user studies – that additional light effects are highly appreciated for both audio and visual contents. Furthermore, [6] includes new research perspectives on ambient intelligence which includes also sensory experiences calling for a scientific framework to capture, measure, quantify, judge, and explain the user experience. Another area that is related to our work is multisensory research (e.g., [7]) which investigates how different senses interact and how their input is integrated to communicate with one another. Finally, [8] provides a good overview of the state-of-the-art in QoE evaluation for multimedia services with a focus on subjective evaluation methods.

In this paper we introduce a slightly different approach to increase the user experience. The motivation behind our work is that the consumption of multimedia assets may stimulate also other senses than vision or audition, e.g., olfaction, mechanoreception, equilibrioception, or thermoception that shall lead to an enhanced, unique user experience. This could be achieved by annotating the media resources with metadata providing so-called sensory effects that steer appropriate devices capable of rendering these effects. This concept is depicted in Fig. 1 and is currently subject to standardization within ISO/MPEG [9]. In particular, the metadata format for describing such sensory effects, i.e., Sensory Effect Description Language (SEDL), will be defined by ISO/MPEG for which we have developed a test-bed enabling the evaluation of Quality of Multimedia Experience (QoMEX). This test-bed is described in this paper as well as preliminary measurements and results. For the preliminary measurements we have developed algorithms that extract sensory effects (i.e., additional light

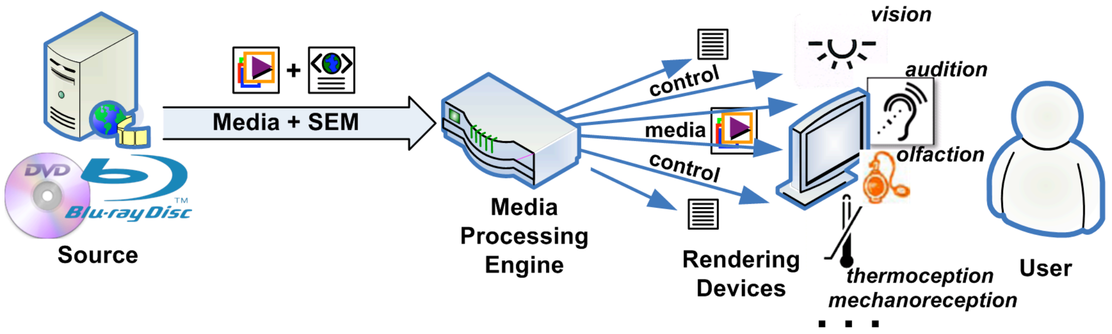

Fig. 1. Concept of Sensory Effects.

effects) directly from the multimedia content with the aim to reduce the metadata description size which should also speed up the authoring process.

The remainder of this paper is organized as follows. An overview of the Sensory Effect Description Language (SEDL) is given in Section 2. The actual test-bed for the Quality of Multimedia Experience (QoMEX) evaluation of sensory effects is described in Section 3 which also provides preliminary measurements and results. Finally, the paper is concluded in Section 4 and future work items are highlighted in Section 5.

2 SENSORY EFFECT DESCRIPTION LANGUAGE

The Sensory Effect Description Language (SEDL) [9] is an XML Schema-based language which enables one to describe so-called sensory effects such as light, wind, fog, vibration, etc. that trigger human senses. The actual sensory effects are not part of SEDL but defined within the Sensory Effect Vocabulary (SEV) for extensibility and flexibility allowing each application domain to define its own sensory effects. A description conforming to SEDL is referred to as Sensory Effect Metadata (SEM) and may be associated to any kind of multimedia content (e.g., movies, music, Web sites, games). The SEM is used to steer sensory devices like fans, vibration chairs, lamps, etc. via an appropriate mediation device in order to increase the experience of the user. That is, in addition to the audio-visual content of, e.g., a movie, the user will also perceive other effects such as the ones described above, giving her/him the sensation of being part of the particular media which shall result in a worthwhile, informative user experience.

The concept of receiving sensory effects in addition to audio/visual content is depicted in Fig. 1. The *media* and the corresponding *SEM* may be obtained from a Digital Versatile Disc (DVD), Blu-ray Disc (BD), or any kind of online service (e.g., download/play or streaming portal). The *media processing engine* acts as the mediation device and is responsible for playing the actual media resource and accompanying sensory effects in a synchronized way based on the user's setup in terms of both media and sensory effect rendering. Therefore, the media processing engine may adapt both the media resource and the SEM according to the capabilities of the various *rendering devices*.

The current syntax and semantics of SEDL are specified in [9]. However, in this paper we provide an EBNF (Extended Backus–Naur Form)-like overview of SEDL due to the lack of space and the verbosity of XML. In the following, the EBNF will be described.

```
SEM ::=[DescriptionMetadata](Declarations|
      GroupOfEffects|Effect|ReferenceEffect)+
```

SEM is the root element which may contain an optional *DescriptionMetadata* followed by choices of *Declarations*, *GroupOfEffects*, *Effect*, and *ReferenceEffect* elements. The *DescriptionMetadata* provides information about the SEM itself (e.g., authoring information) and aliases for classification schemes used throughout the whole description. Therefore, appropriate MPEG-7 description schemes [10] are used, which are not further detailed here.

```
Declarations ::= (GroupOfEffects|Effect|
                 Parameter)+
```

The *Declarations* element is used to define a set of SEDL elements – without instantiating them – for later use in a SEM via an internal reference. In particular, the *Parameter* may be used to define common settings used by several sensory effects similar to variables in programming languages.

```
GroupOfEffects ::=
  timestamp EffectDefinition EffectDefinition
  (EffectDefinition)*
```

A *GroupOfEffects* starts with a *timestamp* which provides information about the point in time when this group of effects should become available for the application. This information can be used for rendering purposes and synchronization with the associated media resource. Therefore, the so-called XML Streaming Instructions as defined in MPEG-21 Digital Item Adaptation [11] have been adopted which offer this functionality. Furthermore, a *GroupOfEffects* shall contain at least two *EffectDefinitions* for which no timestamps are required as they are provided within the enclosing element. The actual *EffectDefinition* comprises all the information pertaining to a single sensory effect.

```
Effect ::= timestamp EffectDefinition
```

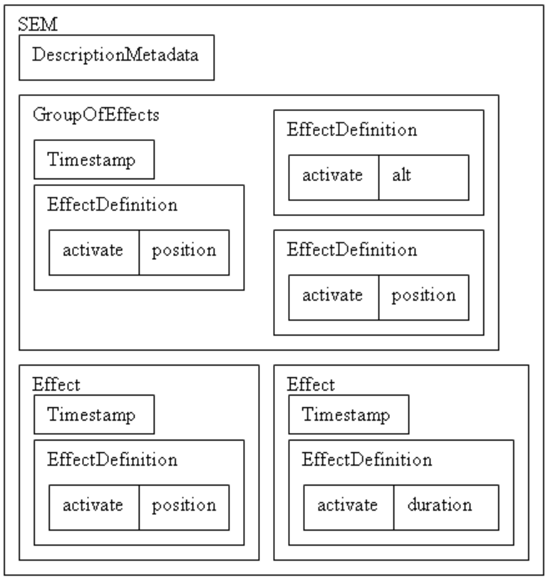

Fig. 2. Abstract illustration of a Sensory Effect Metadata.

An *Effect* is used to describe a single effect with an associated *timestamp*.

```
EffectDefinition ::=
  [activate][duration][fade-in][fade-out]
  [alt][priority][intensity][position]
  [adaptability]
```

An *EffectDefinition* may have several optional attributes which are defined as follows: *activate* describes whether the effect shall be activated; *duration* describes how long the effect shall be activated; *fade-in* and *fade-out* provide means for fading in/out effects respectively; *alt* describes an alternative effect identified by a URI (e.g., in case the original effect cannot be processed); *priority* describes the priority of effects with respect to other effects in the same group of effects; *intensity* indicates the strength of the effect in percent according to a predefined scale/unit (e.g., for wind the Beaufort scale is used); *position* describes the position from where the effect is expected to be received from the user's perspective (i.e., a three-dimensional space is defined in the standard); *adaptability* attributes enable the description of the preferred type of adaptation of the corresponding effect with a given upper and lower bound.

Fig. 2 shows a diagram with the most important entities of a Sensory Effect Metadata.

3 A TEST-BED FOR QOMEX EVALUATION OF SENSORY EFFECTS

In order to annotate media resources with sensory effects and to enable the effect simulation, a test-bed comprising an annotation and simulation tool has been developed which is described in the following. Furthermore, we describe a real-world test environment which can be employed for subjective tests; some preliminary results are given as well.

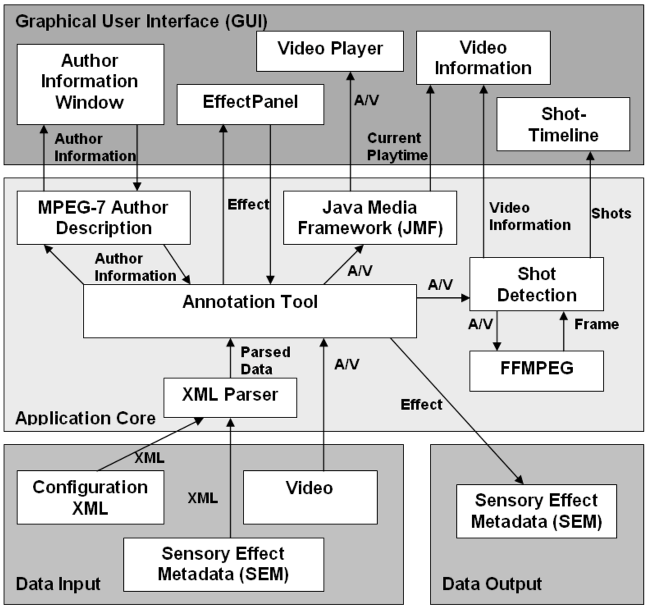

Fig. 3. Architecture of SEVino.

3.1 Annotation Tool – SEVino

The **S**ensory **E**ffect **Vi**deo An**no**tation tool (SEVino) allows for describing various effects for a video sequence based on the constructs and definitions given in [9]. As the standardization process is still in progress, the tool is configurable via an XML document with respect to the available attributes and data types of SEDL and corresponding sensory effects.

SEVino is based on Java and utilizes the Java Media Framework (JMF[1]). JMF is extended via Jffmpeg[2] increasing the number of supported codecs and file formats.

The description of the effects is done via bars which can be drawn like the time bars in a Gantt-diagram. Currently, SEVino does not allow overlapping effects of the same type but there is the possibility to define, e.g., two light types with the same parameter set. The parameter set is used for describing the settings for the different sensory effects.

Entering and editing of sensory effects is done after the bar is created by dragging the mouse to the desired timestamp. The tool displays fields for editing desired settings which are defined in the configuration file.

For providing metadata about the SEM itself (i.e., author, creation time, etc.), SEVino supports the MPEG-7 *DescriptionMetadataType*. The output of SEVino is an XML description compliant to [9]. The timestamps within the SEM description are based on XML Streaming Instructions (XSI) as defined in [11].

Fig. 3 depicts the architecture of SEVino which can be split into three parts. There are data input and data output components for loading already existing SEM descriptions,

[1] http://java.sun.com/javase/technologies/desktop/media/jmf
[2] http://jffmpeg.sourceforge.net

Table 1. Effects supported by SESim.

Effect	Description
Light	Five lights (left, right, three in the background) with RGB support and automatic color calculation.
Wind	Two fans (left, right).
Fog	A fog generator.
Sound	Two speakers (left, right) and a subwoofer with effect support (i.e., echo, hall, etc.).
Vibration	A vibration panel (e.g., a vibration chair).
Temperature	An air-conditioner with range from hot to cold.
Watersprayer	A water-sprayer with interval support.
Shadow	A window blind with closing and opening operation.
Scent	A perfumer with support for different scents (e.g., rose, lilac, etc.).

configuration files for the annotation tool and media resources, which are to be annotated, and for storing generated SEM descriptions.

The central architectural component is the application core which parses the input files and allows filling in the author information based on MPEG-7 (e.g., *mpeg7:Version*, *mpeg7:Creator*, *mpeg7:Name*, etc.). MPEG-7 tags not understood by the tool are ignored during the loading procedure. Furthermore, the application core inserts effects into the effect panel for editing. The effect definitions are read from the configuration document and the existing sensory effects are read from the SEM description in case one has been provided. The media resource is passed to JMF and to the integrated shot detection component. JMF decodes and renders the media resource on the Graphical User Interface (GUI) and it also displays the current playback time on the video information panel. For easy navigation, the annotation tool facilitates a shot detection feature – based on [12] and slightly modified – using a color histogram which selects shot boundaries based on the L1/L2 distance. Because shot detection is resource intensive, every 10th frame is used to determine whether there is a shot boundary. Using this trade-off, the results are acceptable and the shot detection time is moderate on a two core machine. If the L1/L2 distance exceeds a given threshold, the shot is added to the shot timeline in the GUI. Additionally, the shot detection provides information about the video which is displayed in the video information panel.

The GUI is used for presenting the video information and the video. It is also used for editing sensory effects, navigating through the video, and generating the SEM descriptions.

3.2 Simulator – SESim

A **S**ensory **E**ffect **Sim**ulator (SESim) has been developed for the evaluation of the SEM descriptions generated by SEVino. Like SEVino, SESim is written in Java using JMF and Jffmpeg. Currently, the simulator supports nine sensory effects which are summarized in Table 1. For the light effect we have also implemented a simple algorithm that extracts the relevant information directly from the media resource. That is, the average color of the currently displayed frame is used for controlling the light effects. However, as the calculation is CPU intensive we do this only every *n* milliseconds with *n*=500. Note that on a real system this frequency is too low to get an immersive sensation. Furthermore, a real system needs time to activate the corresponding effects via hardware commands which must be taken into consideration also.

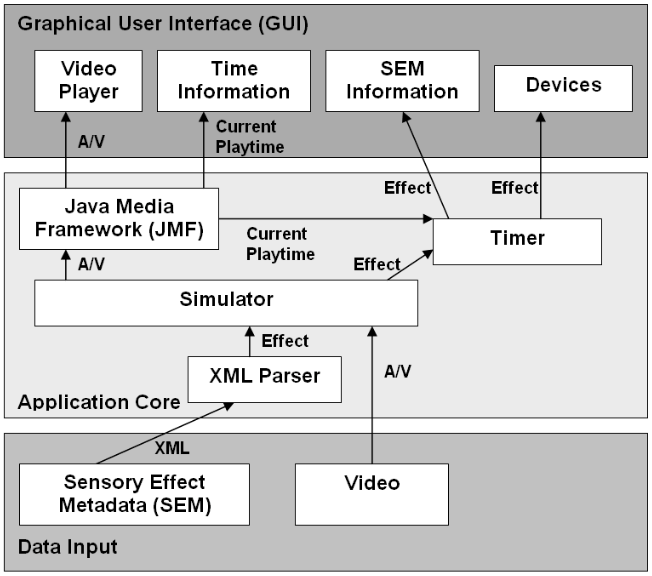

Fig. 4. Architecture of SESim.

Fig. 4 depicts the architecture of SESim. It is divided into three parts which are the input layer, the application core, and the actual GUI. The input layer is used for loading the media resource and the corresponding SEM description which is provided to the application core. An XML parser parses the SEM description and the resulting set of effects is forwarded to a timer. The media resource is passed to the JMF which decodes and renders the media resource on the GUI and it also displays the current playback time. Furthermore, it informs the timer about the current playback time of the media resource such that the timer knows which effect to play. In order words, the timer synchronizes the sensory effects with the actual media resource based on the timestamps provided by both the JMF (i.e., current playback position) and the SEM description. Therefore, the timer extracts the timestamps from the set of effects which correspond to the current playback time of the media resource, displays the settings in the SEM information part of the GUI, and activates the devices which simulate the devices according to their settings. The main purpose of the GUI is to display the media resource and the simulated devices with their effects.

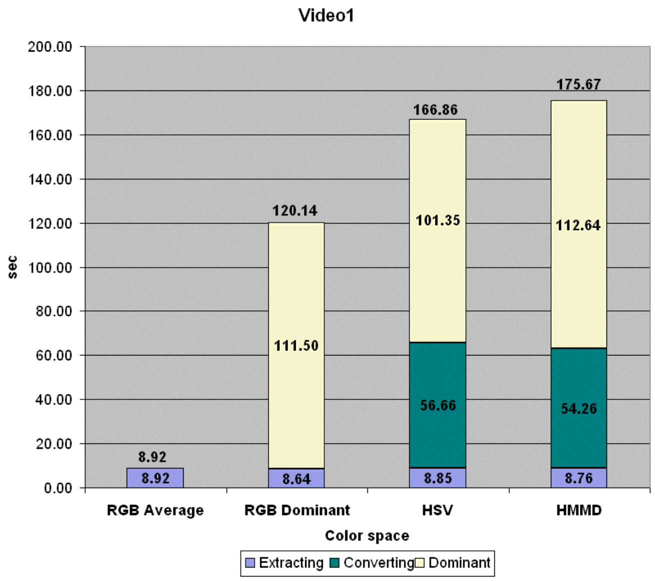

Fig. 6. Performance evaluation for Video1.

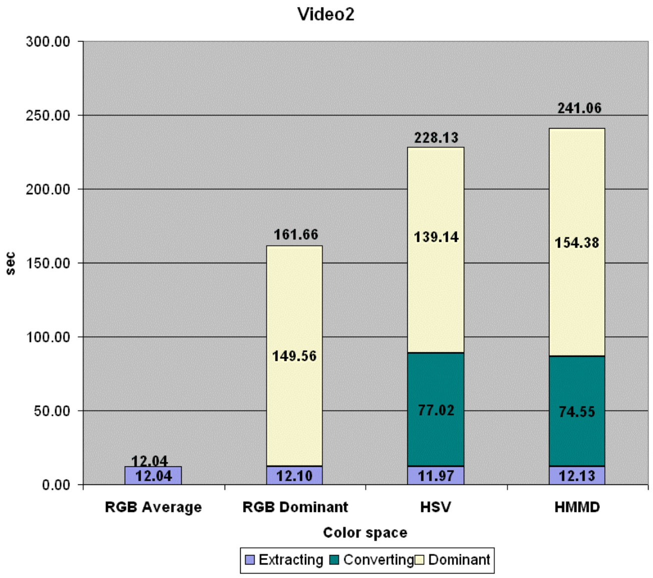

Fig. 5. Performance evaluation for Video2.

3.3 Test Environment

The previous section described a tool which simulates sensory effects and can be used on every computer. In this section a test environment is described which is based on the amBX (Ambient Experience) system [13]. The system consists of two fans, a wrist rumbler, two sound speakers, a subwoofer, two lights and a wall washer.

For the actual test environment we have developed a VideoLAN Client (VLC)[3] plug-in which utilizes the amBX-SDK [13] to control the different devices. The plug-in reads a SEM description and maps the described effects to the corresponding devices and activates the devices at the timestamp given in the SEM description. The light devices are not controlled via the SEM description because within the VLC plug-in an automatic color calculation is deployed. The advantage of the automatic color calculation is that it reduces the description size because light effects do not have to be described explicitly which also speeds up the authoring process. However, different automatic color calculation methods may lead to different user experiences and therefore we have implemented four different algorithms that control the light devices:

(1) **Average color in the RGB** color space: the average color is calculated based on the pixel average of every *n*th pixel with *n* depending on the device type.

(2-4) **Dominant color in the RGB**, **HSV**, and **HMMD** [10] color space: these algorithms use the dominant color according to the RGB, HSV, and HMMD color spaces, respectively.

HSV and HMMD are used since these color spaces are closer to the human perception of color than RGB. The color calculation is done every *m* milliseconds (for *m*=100) which allows for immediate reaction to color changes and results in an intensive sensation. However, the major problem with the color calculation is that it requires a lot of computational resources. In particular, the dominant color algorithm needs much more computational resources than the average color algorithm due to the management of color bins for determining the dominant color for a frame. Please note that the amBX system supports only RGB values which requires additional computational resources due to the back-transformation from HSV/HMMD to RGB.

In the following, we will provide the results of our preliminary measurements for the different automatic color calculation algorithms.

3.4 Preliminary Results

Measurements were done on a Pentium D with 2.8GHz, 1GB RAM and Linux. Video1 (A Chinese Ghost Story 1 - Taoist Monk Fight Scene, http://www.youtube.com/watch?v=TzBkL_1kCUc) has a length of 63 s, 25 fps, 624x336 pixel, and 1058 kbit/s bitrate with a more or less constant color pattern, i.e., the color does not change much. Video2 (Alien Quadrilogy (2003) Trailer, http://www.youtube.com/watch?v=gIWLwen1Rf8) has a length of 62 s, 25 fps, 640x464 pixel, and 702 kbit/s bitrate with a lot of different colors which change very rapidly. Note that the color calculation is performed only on every *p*th frame (*p*=5) for efficiency reasons. The results are shown in Fig. 6 and Fig. 5 and provide the times for the processing of the entire videos. That is, *extracting* refers to the copying of the video frame to an internal data structure, *converting* means the transformation to the respective color space (HSV or HMMD), and *dominant* is referred to as determining the dominant color. As seen from the results,

[3] http://www.videolan.org

only the average color calculation is qualified for real-time extraction. Dominant color in the RGB color space is not applicable for real-time extraction due to the time consumption for determining the dominant color with high resolution (i.e., 24 bit). For HSV and HMMD we observe the same behavior, i.e., the bottleneck is in the management of the different color bins which requires a lot of computational and memory resources. This makes these techniques infeasible for real-time extraction.

4 CONCLUSION

In this paper we have described a test-bed for the QoMEX evaluation of sensory effects consisting of SEVino (i.e., a video annotation tool for sensory effects), SESim (i.e., a corresponding simulation tool), and a real world test environment based on the amBX system and SDK. Furthermore, we have presented preliminary measurements and results. The major findings of this paper can be summarized as follows.

Using the average color for the automatic color calculation enables an immediate reaction to color changes in the content resulting in appealing effects with low computational requirements. Thus, the average color algorithm is suitable for real time extraction which can be further improved by using only every *n*th pixel for calculating average color, e.g., for low-end devices like set-top boxes.

The HSV and HMMD dominant color algorithms provide a smoother reaction to color changes in the content but have higher computational requirements. Therefore, real-time extraction is not achievable on low-end devices and, thus, additional metadata support would be required. That is, the color information is not extracted from the media resource but provided as metadata either within the sensory effect metadata or as, e.g., MPEG-7 description.

Finally, the different schemes for automatic color calculation could be implemented as different levels (e.g., level one uses average color, level two HSV dominant color, etc.) to be selected depending on the users' preferences or characteristics (e.g., age, mood, handicaps).

5 FUTURE WORK ITEMS

The future work items in this area can be clustered into three major parts. First, we will further optimize the automatic color calculation with a special emphasis on real-time support and integration into VLC. Second, we will perform subjective tests with the aim to derive new quality metrics for the perception of media resources that are enriched with sensory effects. Finally, we will investigate means for (semi-)automatic extraction of sensory effect information – beyond light effects – from the associated media resource using both image/video and audio analysis tools.

6 REFERENCES

[1] A. Vetro, C. Christopoulos, T. Ebrahimi (eds.), *Special Issue on Universal Multimedia Access*, IEEE Signal Processing Magazine, vol. 20, no. 2, March 2003.

[2] F. Pereira, I. Burnett, "Universal Multimedia Experiences for Tomorrow," *IEEE Signal Processing Magazine*, vol. 20, no. 2, March 2003, pp. 63-73.

[3] D.S. Hands, "A Basic Multimedia Quality Model", *IEEE Transactions on Multimedia*, vol. 6, no. 6, December 2004, pp. 806–816.

[4] F. Pereira, "A triple user characterization model for video adaptation and quality of experience evaluation," *Proceedings of the 7th Workshop on Multimedia Signal Processing (MMSP) 2005*, Shanghai, China, October 2005, pp. 1–4.

[5] B. de Ruyter, E. Aarts. "Ambient intelligence: visualizing the future", *Proceedings of the Working Conference on Advanced Visual Interfaces*, New York, NY, USA, 2004, pp. 203–208.

[6] E. Aarts, B. de Ruyter, "New research perspectives on Ambient Intelligence", *Journal of Ambient Intelligence and Smart Environments, IOS Press*, vol. 1, no. 1, 2009, pp. 5–14.

[7] The International Multisensory Research Forum (IMRF), http://www.imrf.info/. (last accessed: May 2009)

[8] M. Grega, L. Janowski, M. Leszczuk, P. Romaniak, Z. Papir, "Quality of Experience Evaluation for Multimedia Services", *Przegląd Telekomunikacyjny*, vol. 81, no. 4, 2008, pp. 142–153.

[9] C. Timmerer, S. Hasegawa, (eds.) "Working Draft of ISO/IEC 23005 Sensory Information," *ISO/IEC JTC 1/SC 29/WG 11/N10475*, Lausanne, Switzerland, February 2009.

[10] B. S. Manjunath et al., *Introduction to MPEG-7: Multimedia Content Description Interface*, John Wiley and Sons Ltd., June 2002.

[11] ISO/IEC 21000-7:2007, *Information technology - Multimedia framework (MPEG-21) - Part 7: Digital Item Adaptation*, November 2007.

[12] M. Lux, S.A. Chatzichristofis, "Lire: Lucene image retrieval: an extensible Java CBIR library", *Proceeding of the 16th ACM International Conference on Multimedia*, Vancouver, Canada, October 2008, pp. 1085–1088.

[13] amBX, http://www.ambx.com. (last accessed: May 2009)